Too Big to Cheat : Mining Pools' Incentives to Double Spend in Blockchain Based Cryptocurrencies

Soria Ruiz-Ogarrio, Jorge

2019-06-11

Soria Ruiz-Ogarrio , J & Savolainen , V 2019 , ' Too Big to Cheat : Mining Pools' Incentives to Double Spend in Blockchain Based Cryptocurrencies ' , Paper presented at GSF and FDPE Summer Workshop in Finance , Jyväskylä , Finland , 10/06/2019 - 10/06/2019 .

http://hdl.handle.net/10138/309233

unspecified
submittedVersion

Too Big to Cheat: Mining Pools' Incentives to Double Spend in Blockchain Based Cryptocurrencies

Ville Savolainen * Jorge Soria Ruiz-Ogarrio†

December 19, 2019

Abstract

In most blockchain based cryptocurrencies majority of verification power is required for facilitating a successful double spending attack, i.e. using the same funds multiple times. Because possibility to double spend sharply deteriorates trust and value, concentration is traditionally considered to be a significant problem. We model agents' incentives to facilitate double spending attacks under opportunity costs. Contrary to a host of previous literature, our main findings indicate that under meager economic profits large miners have higher incentives to act honestly than outsiders even in the absence of any direct costs. Intuitively, mining pools holding substantial amounts of power in a cryptocurrency also have much to lose if the value collapses.

Keywords: Blockchain, Cryptocurrencies, Bitcoin

JEL Codes: D43, E42, G29

*Hanken School of Economics; ville.savolainen@hanken.fi +358451365687

†University of Helsinki, Helsinki Graduate School of Economics; jorgesro@gmail.com

1 Introduction

The first implementations of electronic cash[1] required a trusted third party to verify transactions and prevent double spending. In a double spending attack, the very same virtual money is used multiple times and sellers are left without a payment. Because no seller would be willing to deliver goods if the payments were not immutable, these attacks are considered to be a major threat to crypto-currencies' viability. Nakamoto [Nakamoto, 2008] introduced a novel decentralized double-spending preventing transaction verification solution - blockchain technology[2]. In which records are stored in blocks which are cryptocraphically chained and held by multiple nodes in a peer-to-peer network. To cancel a payment after receiving goodsa fraudulent agent has to tamper the blockchain by redoing a costly proof-of-work (PoW). To succeed, a malevolent actor has to outpace all of the network's honest agents, which requires controlling more than 50 percent of network's computational power. Consequently, concentration of computational power is considered to open a door for double spending attacks.

Value appreciation of cryptocurrencies has increased popularity of mining, which is defined as using processing power (i.e. hashing power) to generate new blocks. Increased number of miners has decreased the probability of an individual miner to mine a new block and, therefore, to win the rewards. Consequently, small scale mining has become very risky. Therefore, as a natural hedge against idiosyncratic risk, most miners have joined mining pools. Pools have acquired such popularity that since 2015 between 95 and almost 100 percent of the hash (processing) power in Bitcoin has been controlled by pools. The situation

[1]The idea of cryptographic currencies was -most probably- first proposed by Chaum [Chaum, 1982]. During the 90s and through out the expansion of the Internet, the idea of digital money was latent in various fields. Some early commercial attempts to develop cryptographic protocols for e-money were e.g. Beenz, DigiCash, Flooz and Peppercoin [Bonneau et al., 2015] [Huang, 2003]. The milestones in the process include launching e-gold Ltd. in 1996 and PayPal in 1999. E-gold was the initially successful and widely accepted digital currency, however due to multiple legal issues, it was ultimately shut down. Although PayPal was introduced when e-gold was already relatively established in the market, it managed to gain the attention of successful companies, such as Elon Musk's X.com and, latter, eBay. This could explain its fast and durable success.

[2]An idea similar to blockchain had been proposed by Haber and Stornetta in 1990 [Haber and Stornetta, 1990].

is similar in all major crypto-currencies.

Concentration of verification process to few large pools, especially in the major cryptocurrencies Bitcoin and Ethereum, has been a remarkable concern especially amongst computer scientists and practitioners. Gencer et al. present empirical results concerning the distribution of mining power in Bitcoin and Ethereum networks: During 2015-2017 top four mining pools controlled 51 percent of the mining power in Bitocoin, and in Ethereum the three largest pools alone controlled 63 percent of the mining power on average. A better Byzantine fault tolerance[3] could be acquired by employing 20 autonomous desktop computers [Gencer et al., 2018]. Furthermore, in Bitcoin some mining pools have controlled over 50 percent of hashing power: DeepBit (June 2011), BTC Guild (April 2013) and Ghash.io (July 2014). Currently the largest mining gear producer Bitmain Technologies Ltd controls AntPool and BTC.com pools which combined have around 30 to 40 percent of Bitcoin total hashing power. In addition, Bitmain has large proprietary mining operations. The situation is frequently more severe amongst crypto-currencies with small market capitalization. These findings are concerning in the light of Nakamoto's original idea [Nakamoto, 2008]. He famously argues that the security of a cryptocurrency rests on its decentralization. Furthermore, Miller and LaViola – among others – demonstrate that proof-of-work in Bitcoin protects the protocol from fraudulent nodes, as long as the majority of the processing power of the network is controlled by nodes that follow correctly the protocol [Miller and LaViola Jr, 2014].

However, from an economist's perspective centralization is not necessarily a problem whenever agents' increased capability to conduct double spending attacks is countered by increasing incentives to maintain honest conduct. Given the concerning levels of pool concentration, we build our model to better understand large pools' incentives for maintaining

[3]A central topic in decentralized protocols is how to achieve consensus among peers in a network, especially when nodes cannot differentiate trustworthy nodes from fraudulent ones. Blockchain protocols are designed to have a high degree of Byzantine fault tolerance (BFT), meaning that the protocol can remain trustworthy even if multiple nodes fail or send fraudulent information. The academic discussion about the topic began with [Pease et al., 1980].

honest conduct. The contribution of this paper is to demonstrate that centralization is harmless. To our knowledge, this paper is the first to formally present such result. Intuitively, larger pools would suffer larger loses if the value of a cryptocurrency is deteriorated by a decay in trust caused by a significant double spending attack. In our analysis we concentrate on proof-of-work protocol because it is the most widely used verification protocol. However, the same economic intuition is applicable to other verification protocols[4]. We choose to limit our analysis to the double spending attack, because of its relevance for cryptocurrency users (i.e. buyers and sellers exchanging goods in a cryptocurrency). Other attacks are mainly conducted by miners against pools or large miners or pools against smaller pools or miners, with the exception of sabotage Goldfinger attacks that aim at destroying the whole cryptocurrency for exogenous reasons [Budish, 2018].

We cast our model in an industrial organization setting, resembling Green and Porter [Green and Porter, 1984]. In our model miners choose to hedge against idiosyncratic risk by conducting mining in pools, this is corroborated by empirical observations. Because of entry costs, incumbent pools gain economic profits in a collusive equilibrium. Pools' discounted future profits are increasing in pool size and hence the opportunity cost for fraudulent behavior is increasing in pool size. We derive a threshold for which for each pool cost of double spending is higher than an outsider's cost. In our model, the results hold for pool fees and profits 100-1000 times smaller than those common in the industry.

Anecdotally, in May 2019 two largest pools in Bitcoin Cash conducted a protective action by jointly conducting a majority attack to reverse a malevolent transaction. Hence, some pools do have incentives and capability for not only acting honestly but also maintaining trust in the network. Probably a more well-known case is the 2017 split of Ethereum to Ethereum Classic and Ethereum in which some miners decided to cancel malevolent transactions and forked the protocol. Those opposing the cancellation continued with the chain that contained

[4]In proof-of-stake protocol distributed consensus is achieved by randomly selecting a creator of the next block from those holding a stake (current wealth and possibly how long the wealth was held). Stake holders may participate in a pool and hence receive rewards more regularly. The model presented should be applicable to a pool operating in a proof-of-stake cryptocurrency.

those transactions and called the fork Ethereum Classic, while those whom wanted to cancel the fraudulent transactions forked to another chain that kept the name Ethereum. Currently the market capitalization of Ethereum is about 40 times higher than that of Ethereum Classic. This latter example further illustrates that large pools have strong incentives to keep the network safe and trustworthy.

Our results' external validity relies on the assumption that pools make economic profits. Unfortunately, sufficient data about profitability of pools is practically nonexistent. As an exception, Bitmain the company owning and operating two largest Bitcoin mining pools with 30 to 40 percent of the hashing power reported 36.3 million dollar gross profit from mining pool service for the first six months of 2018 with a gross profit margin of 84 percent in its Proof of Application filing to Hong Kong Security Exchange in 2018. Since 2015 the margins have ranged from 80 to 89 percent [BitMain, 2018]. In addition, two empirical findings support the existence of economic profits: First, larger pools win mining competitions with larger probability than their relative computational power would suggest [Gencer et al., 2018]. Hence, even under constant (or decreasing) scale costs, in Bertrand type competition, largest pool(s) would generate economic profits due to higher quality. Second, observed pool fees are increasing in pool size [Cong et al., 2018]. Cong et al. propose that this is due to the large pools' market power generated by frictions in miners' willingness to change pools.

We focus on monetary incentives and hence limit our scope to exclude other important aspects of decentralization such as geographic concentration which might have political implications[Kaiser et al., 2018], project competition [Van Wirdum, 2018], different attacks against smaller miners, cryptocurrency holders' taste for diversification, inequality aversion etc. Furthermore, for our purposes, an agnostic view about why pools sizes are heterogeneous and what causes the empirically observed concentration is sufficient. Whereas, [Cong et al., 2018] provide valuable insights on how pool fees are set and how miners allocate their mining power to different pools and hence why pool sizes differ and vary over

time[5].

To our knowledge, our paper is the first to propose conditions for pool concentration not to be harmful. Hence, our work directly relates to [Nakamoto, 2008] and proposes a novel and contradicting perspective about the importance of decentralization.

2 Literature Review

Our paper is closely related to papers discussing blockchains' resilience against double spending attacks. [Budish, 2018]; [Kroll et al., 2013]; and [Rosenfeld, 2014] discuss double spending attacks as a threat to trust in a blockchain. [Kroll et al., 2013] argue that whenever there exists a possibility to double spend, because sellers cannot differentiate between honest and fraudulent buyers, sellers should cease to accept transactions. Hence, leading to a collapse in cryptocurrency's value. [Budish, 2018] builds a model to address double spending and draws insights regarding miners' incentives, fixed and variable mining costs, and double spending. [Rosenfeld, 2014] is one of the earliest academic works to discuss double spending and motives behind it. Both Budish and Rosenfeld acknowledge that a double spending attack could harm the double spender, if she has a stake in the cryptocurrency. Our model contributes by analysing pools and their incentives to double spend.

Chiu and Koeppl discuss the tradeoff between fast transactions and finality of payments in a proof-of-work protocols [Chiu and Koeppl, 2019]. Eyal and Sirer discuss a different type of incentive compatible attacks, conducted by large miners or pools, where larger entities profit more per hash than smaller entities [Eyal and Gun Sirer, 2013]. Kiayias, Kotsoupias, Kyropoulou, and Tselekounis present similar results [Kiayias et al., 2016]. Paganotta and Buraschi discuss aggregate mining power as a factor increasing trust worthiness and hence value [Pagnotta and Buraschi, 2018]. Research more loosely related to our work discussing blockchain from the perspective of market structure (or design) of mining activity is ex-

[5]They also propose that there are economic forces limiting pool growth and hence they provide much needed economic illumination to blockchain discussion.

tent [Gans and Halaburda, 2015][Böhme et al., 2015] [Huberman et al., 2017][Dimitri, 2017] [Ma et al., 2018][Cong et al., 2018][Biais et al., 2018][Easley et al., 2019]. Gans and Halaburda discuss competition between cryptocurrencies. Cong and He discuss blockchain and smart contracts in an industrial organization setting [Cong and He, 2019]. Yermack discusses blockchain from a perspective of corporate governance and provides a detailed introduction [Yermack, 2017]. Lee discusses securities trading in a blockchain [Lee, 2016].

3 Model

To better understand how concentration in a blockchain affects double spending attacks we consider pools and miners in an industrial organization framework. We find that concentration in mining power is harmless for the networks resilience against double spending attacks. The findings stem from the fact that, the larger a pool is, the more it loses if the network value collapses. Hence, even if a large pool is more able to conduct mischief, it should be less willing to do so. Our model is stylized, yet its intuition carries over to other settings where large miners, pools or coalitions receive economic profits.

3.1 Model Setup

Consider a world in which time is infinite and discrete and is indexed by t, $t = 0, 1, 2, \ldots$ There are two types of agents – miners and pools – having a discount factor $\beta \in (0, 1)$. Miners are homogeneous, risk averse and atomistic, whereas pools are risk neutral. In every period $t \geq 0$, miners choose their hashing power at a unit cost C, and hashing power allocation h_m for each pool $m \in \{1, 2, ..., M\}$ and h_o for solo mining.

3.2 Mining Pools

Mining pools offer different fee and reward contracts; the simplest mechanisms being proportional payment and pay-per-share[6]. In a proportional reward system[7], whenever a pool wins a mining competition a miner receives

$$(1 - f^m)R\frac{h_i}{H_m} \tag{1}$$

where h_i is the miner's hash rate contributed to the pool m, R is block reward, H_m is the total hashing power in that pool and f^m is a fee collected by pool m. In a pay-per-share reward mechanism a pool effectively rents miner's hashing power and pays a rent regardless of whether the pool wins block rewards or not, fully insuring participating miners. However, pay-per-share is uncommon and usually associated with significantly higher fees. In addition, diversification of miner's hashing power to different pools would effectively insure miners against idiosyncratic risk. Hence, in our model we choose to concentrate on proportional reward mechanisms.

3.2.1 Collusive Equilibria

We restrict each pool's strategy to the standard supergame *grim trigger strategy*. Specifically, consider the following strategy for M incumbent pools to collude:

1. *Collusion*: In every period, pools agree upon a fee f^c. Miners allocate their hashing power to pools.

2. *Punishment phase*: once one of the incumbent pools does not have any participants,

[6]For more a detailed description of the most common fee and reward mechanisms used by pools, see Appendix A.

[7]The simplicity of proportional reward makes it vulnerable to block withholding and pool-hopping attacks, where a miner receives more than her proportional share of the rewards. Hence, more complicated reward mechanisms have been developed. The purpose of these structures is to ensure that miners receive rewards proportional to the hashing power they have contributed and avert other malicious behavior [Rosenfeld, 2011]. Assuming that these means are effective we can, without a loss of generality, model all pools' reward mechanisms as proportional, where miners in all pools receive $(1 - f^m)R\frac{h_i}{H_m}$ whenever a pool wins a mining competition.

punishment phase is triggered and the pools enter into a Bertrand competition. In absence of marginal costs, and because the pools are homogeneous, the pools will receive zero profits.

In a collusive phase the pools discounted future profits are

$$\sum_{t=0}^{\infty} \beta^t \frac{H_m}{H} f^c R = \frac{f^c R}{1-\beta} \frac{H_m}{H} \tag{2}$$

where H is network's total hashing power and β is the time discount factor.

Corollary 1 *A collusive strategy is an equilibrium if*

$$\sum_{t=0}^{\infty} \beta^t \frac{H_m}{H} f^c R = \frac{f^c R}{1-\beta} \frac{H_m}{H} > f^c R \qquad \forall \quad \frac{H_m}{H} \tag{3}$$

Corollary 1 states that the profit from lowering the fee, and hence capturing the whole market, should be less than the value of discounted future profits in collusion phase. From this naturally follows:

Proposition 1 *If*

$$\exists \frac{H_m}{H} \qquad for \quad which \qquad \frac{H_m}{H} < \ 1-\beta \tag{4}$$

no collusion equilibrium exists.

Proposition 1 states that – given the discount factor – there should not exist extremely small pools for collusion equilibrium to exist. E.g. for annual β of 0.9, there should exist pools vesting less than 0.0002 percent of hashing power for collusion strategy not to be a Nash Equilibrium. For the remaining part of the analysis we will assume that such pools don't exist in the market, thus a collusion equilibrium can be sustained.

Above, we have assumed that R is constant. In reality, because rewards are paid in a cryptocurrency, they are highly volatile. In our model this would yield the same result, because pools are assumed to be risk neutral. In addition, (some) crypto-currencies

have expected declines in rewards (e.g. BTC reward is halved every 210,000 blocks, which occurs approximately every four years). It is a standard result that in these cases the benefit from deviating would be highest just prior to the expected decrease in reward [Rotemberg and Saloner, 1986]. For parsimony we have restricted our analysis from considering such cases.

3.2.2 Entry and Collusive Fee Setting

Every period $t \geq 0$ there exists a possible entrant pool without miners. Therefore, an entrant would set a fee $f^e < f^c$ to obtain miners. Prior to an entry the entrant pays a positive entry fee ζ. An entry will trigger the price competition phase and, hence, each pool makes zero profits post entry.

Therefore, a condition for a feasible entry is given by

$$f^e R - \zeta > 0 \quad \text{where} \quad f^e < f^c \tag{5}$$

Corollary 2 *It follows from feasible entry condition (Equation 5) that in order to deter entry colluding pools set a fee* f^c

$$f^c \leq \frac{\zeta}{R} \tag{6}$$

To keep the model parsimonious we have chosen a very simple barrier of entry as is manifested by Corollary 2. However, one could equivalently assume that, once an entry occurs only an active fraction of miners observes it. Hence the active miners would face a trade-off between lower fees and smaller diversification benefits. In this case, to deter entry incumbent pools' fee setting strategy should make active miners indifferent between choosing an entrant pool or staying in incumbent pools. In addition, incumbent pools have likely established credibility for not siphoning rewards, having a reliable infrastructure etc. all attributes that an entrant might easily lack.

3.3 Miners

In every period t, a reward R is randomly assigned to a solo miner or a pool[8]. The probability of winning the reward in every period t is $\frac{h_i}{H}$ for a solo miner and $\frac{H_m}{H}$ for a pool, where H is network's total hash power and H_m is pool m's hashing power. Whenever a pool wins the mining competition it collects a fee $f^m \in (0,1)$ and distributes the remaining reward to participants according to their contribution to the pool's total hashing power $\frac{h_i}{H_m}$. The miner j's expected utility at t for $t+1$ is hence given by the von-Neumann-Morgenstern Utility Function

$$U(H_j) = \frac{h_0}{H} u\left(R - C\sum_{i=0}^{M} h_i\right) + \sum_{i=1}^{M} \frac{H_m}{H} u\left((1-f^m)R\frac{h_i}{H_m} - C\sum_{i=0}^{M} h_i\right) \tag{7}$$

where, $U(\ )$ is a continuous, monotonic and concave utility function and h_0 is the allocation to solo mining and h_m $m \in [1,2,3,\ldots,M]$ are the allocations to M different pools. Each pool sets a fee f^m to maximize its profit.

3.3.1 Equilibrium Hashing Power and Allocation

Proposition 2 *Given fees and total hashing power, all miners' symmetric allocations among pools offering the lowest fee are Subgame Perfect Equilibria.*

Proposition 2 was initially discussed by Cong et al [Cong et al., 2018]. Following intuition of Modigliani-Miller [Modigliani and Miller, 1958] the initial pool size does not matter whenever miner's are able to diversify by allocating their hashing power to multiple pools. Hence, any allocation **where all pools get a share and?** that is symmetric amongst the miners is a Nash Equilibrium. By a symmetric allocation we refer to an allocation in which each miner j allocates the same proportion of hashing power as all the other miners to each pool i.e. $\frac{h_{m,j}}{H_j} = \frac{h_{m,-j}}{H_{-j}}$ for each $m \in [1,2,3,\ldots,M]$ and j.

Corollary 3 *Miners allocate their hashing power amongst pools.*

[8]see Appendix C for more detailed discussion.

To acquire miners, pools set fees for which miners prefer pools over solo mining. If miners are atomistic, once a miner prefers mining in pool(s) over solo mining all miners will prefer pools over solo mining. Because pools, in our model, do not have costs and miners are risk averse, there exists a fee $f^m > 0$ for which miners prefer pools and which pools are willing to offer.

Proposition 3 *Miners' utility function simplifies to the Bernoulli utility function*

$$U(H_i) = u\left((1-f^c)R\frac{\sum_1^M h_i}{H} - C\sum_1^M h_i\right) = 0 \tag{8}$$

Proof. It follows from the assumptions that miners are atomistic and mining is competitive, that miners gain zero utility in equilibrium. Therefore, by employing Proposition 2 and Corollary 3 we get Proposition 3. ■

By allocating according to Proposition 2 miners are able to perfectly diversify mining risk. Total costs are equivalent to a net reward payed to miners. Hence, profits for miners are zero. This simplifies our analysis and corresponds to what is observed in most crypto-currencies, namely that small scale mining is not profitable.

As proposed above, all miners symmetric allocations are Nash Equilibria. Miners, however, would need to coordinate to reach this allocation. Hence, to simplify our analysis we make the following assumption:

Assumption 1 *Miners coordinate their allocation amongst pools offering lowest fees at t by employing aggregate allocation at $t-1$ as a focal point in every period $t > 0$. Miners' allocation at $t = 0$ is exogenously given.*

In the absence of a definite coordination device, a focal point may function as such [Schelling, 1960][Mehta et al., 1994][Bacharach and Bernasconi, 1997]. We argue that if a set of pools is homogeneous and provides the same service for the same price, previous aggregate allocation is a natural focal point for miners to allocate hashing power. This is

accentuated, when there exists a large number of miners causing coordination to be unfeasible. An allocation determined by a focal point is an allocation in the set of possible Nash Equilibria allocations given by Proposition 2. The assumption implies that, ceteris paribus, pool sizes are stable[9].

Proposition 4 *In equilibrium total hashing power H is a function of f, R and C*

$$H = \frac{(1 - f^c)R}{C} \tag{9}$$

Proposition 4 follows from Proposition 3 by summing over all miners and it states that in equilibrium, because miners are fully insured against idiosyncratic shocks and make zero profits, total cost of hashing power equals the net reward.

3.4 Blockchain Security and Concentration

The purpose of proof-of-work (PoW)[10] is to make block generation expensive to avoid spam. Most cryptocurrency protocols follow the longest chain rule, in which blocks are linked to the chain that has the most blocks and hence the most proof-of-work invested in it. Proof-of-work and mining rewards[11] ensure that block-mining requires large amounts of processing power[12].

[9]In reality pool sizes vary, however, as long as the current pool size is the best predictor of the next period pool size this does not affect our results.

[10]Early work on proof-of-work is difficult to track down in history. However, the term and its first formalization can be found in [Jakobsson and Juels, 1999], where authors propose the name Proof-of-Work for an idea already present in various works. Jakosson and Juels attribute the first conceptualization of PoW to Dwork and Naor [Dwork and Naor, 1992], who present an application to prevent spam by forcing the use of processing power to generate a cost to mail sending and thus to deter junk mail. Without knowing the contribution by Dwork and Naor, Back suggested a similar idea in 1997 `www.hashcash.org/papers/announce.txt,1997`. In 2002 paper he later acknowledged the similarities between both ideas [Back et al., 2002]. More detailed information about proof-of-work can be found in [Becker et al., 2012] and [Laurie, 2004].

[11]e.g. currently in Bitcoin the miner of a block receives a reward of 12.5 bitcoins, which are created with the new block. The amount halves every 210,000 blocks, so that the supply of bitcoins per block decreases over time until reaching total supply of 21 million Bitcoins

[12]For example, in the case of Bitcoin from a global hash rates going under 0,15 TH/s in the beginning of 2011 to values fluctuating around 50 000 000 TH/s during the first quarter of 2019. Energy wise, if Bitcoin were a country, it would rank around 40th in energy consumption per year, slightly behind Chile.

Because new blocks are added on top of an exiting block[13], the probability of successfully mining a longer alternative chain starting from that block decreases exponentially with the number of blocks mined, unless an attacker holds more than 50% of the hashing power. Consequently, discussion on the security of open blockchain protocols is often focused on majority attacks.

Controlling over a half of the hashing power makes it technically feasible to fraudulently alter the chain. However, as will be demonstrated below, a party controlling enough hashing power seldom has incentives to conduct such attacks. A double spending attack may be facilitated without any direct cost by a pool controlling more than a half of the network's hashing power. This has been considered one of the main problems of pool concentration. The attack may also be facilitated by an entity controlling less than a half, if it acquires the needed hashing power.

3.4.1 Cryptocurrency's Value Post Successful Double Spending Attack

Once sellers observe a double spending attack they can infer that someone has a capacity to conduct such attacks and these attacks are in those agents best interest. If the sellers could differentiate between those capable and incapable of conducting double spending attacks they could choose their customers accordingly. However, because blockchain protocols are anonymous by nature this is not achievable. Hence, the sellers would need to set a premium for payments made in that cryptocurrency. This however, would lead to devaluation of the currency and to a further need to increase prices. Without major changes in the protocol design, this would eventually lead to a collapse in the value. The situation is reminiscent of

[13]However, it is not necessarily incentive compatible to follow the rule: Kroll, Davey and Felten [Kroll et al., 2013] argue that mining the longest chain is one possible Nash equilibrium, perpetuated mainly because it acts as a focal point. However, there exist infinitely many alternative equilibria, which could result if a large enough actor or coalition impels them. Biais, Bisière, Bouvard and Casamatta [Biais et al., 2018] demonstrate that although miners usually find it optimal to cooperate, there are situations where coordination is sub-optimal causing a blockchain to fork into two or more alternative main chains (e.g. Ethereum Ethereum Classic fork discussed above). Furthermore, also [Yermack, 2017] expresses concerns regarding the use of both public and private blockchains. He elaborates that even decentralized public chains are subject to changes *ex post*, if sufficient proportion of chain's members join to undo some outcome.

Akerlof's lemons problem [Akerlof, 1970], although what in our case destroys the market is the information asymmetry on the value of the payment rather than uncertainty about the value of the purchased good. Hence, we make the following assumption

Assumption 2 *Once a double spending attack is successfully conducted, trust in the network vanishes and value of the future rewards decreases to zero.*

The outcome would be the same if we assumed that only a fraction of value is lost due to an attack. Because it would be an optimal strategy to attack subsequently, rational agents would infer from one attack that the there will be a series of attacks, eventually driving the value of the currency to zero. Therefore, through agents' anticipation, the value of rewards would decrease to zero once an attack has taken place.

3.4.2 Cost of Conducting Double Spending Attack

An agent controlling any positive amount of hashing power may attempt to conduct a double spending attack, and will succeed with some strictly positive probability. Hence, if the pools hashing power does not decrease due to an attack attempt (and attacking is free), a pool could conduct an infinite number of attempts and would therefore with certainty conduct a successful double spending attack for free for any positive amount of controlled hashing power. In reality, after a while, pool members would detect that the pool is trying to conduct a double spending attack (or siphoning rewards), because pools mining forked chains do not receive rewards in the main chain and hence can not reward its members. To avoid such an obscure result we postulate:

Assumption 3 *Once commenced a double spending attack cannot be canceled and it has to conclude at some arbitrary time T.*

To conduct a double spending attack an agent has to control a sufficient amount of hashing power. If a miner lacks it, she may increase her capacity buy acquiring required

facilities, mining gear or by renting hashing power from other miners. These two options are analogous to buying computer storage or renting it from a cloud service provider. For the rest of the paper we will assume that the market for hashing power is frictionless.

Assumption 4 *Market for hashing power is frictionless.*

An attacker has to wait for a certain number of blocks K to be mined in the main chain before the seller releases the goods. After this, the attacker broadcasts a chain which is longer than the main chain and hence becomes the new valid chain. The attacker chooses hashing power $\frac{H_a+H_m}{H}$ to minimize the expected cost of conducting a successful attack, where H_a is the hashing power that an attacker acquires in addition to H_m which is the hashing power already controlled by the attacker.

There exists two conditions for a successful attack:

1. The main chain has reached the block where goods are released.

2. The attacker's chain must be one block longer than the main chain.

Implying that $B_{attack} > B_{main} \geq K$ where B_{attack} is the number of blocks mined by the attacker, B_{main} the number of blocks added to the main chain starting from the block including the fraudulent transaction and K is the number of escrow blocks starting from the transaction block.

Corollary 4 *In a frictionless market, once the main chain has reached a length where goods are delivered, an attacker will acquire all available hashing power and then broadcast a chain that is one block longer with a cost of* $(K + 1 - B_{attack})\left(1 - \frac{H_m}{H}\right)$.

See Appendix D for a proof.

Corollary 5 *While* $B_{main} < K$ *an attacker's hashing power is* $\frac{H_m}{H}$.

An attack may conclude in two manners: First, an attacker has mined $K+1$ blocks before $B_{main} = K$ and waits before broadcasting the new blocks. Second, $B_{main} = K$ and an attacker mines $B_{main} + 1 - B_{attack}$ blocks. While $B_{main} < K$ an attacker has a strictly positive probability of mining enough blocks for a successful attack without any cost. Hence, by increasing hashing power over $\frac{H_m}{H}$ an attackers increases the probability of "wasting" free hashing power.

See Appendix D for a proof.

By combining Corollary 4 and 5, the expected direct cost for conducting a double spending attack can be expressed as

$$\sum_{a=0}^{K} \underbrace{\left(1-\frac{H_m}{H}\right)^K \left(\frac{H_m}{H}\right)^a \binom{K+a}{a}}_{\text{Probability mass function Pr(X=K)}} \underbrace{\left(1-\frac{H_m}{H}\right)(K+1-a)CH}_{\text{Cost for each a}} \tag{10}$$

If we substitute CH with the equilibrium condition $CH = (1-f^c)R$ given in Proposition 4, and consider the opportunity cost $\frac{H_m}{H}\frac{f^cR}{1-\beta}$ (Equation 2) from losing the future profits, we get the following Theorem:

Theorem 6 *In a frictionless markets, the expected cost for conducting a double spending attack for* $\frac{H_m}{H} \leq \frac{1}{2}$ *is*

$$\sum_{a=0}^{K} \underbrace{\left(1-\frac{H_m}{H}\right)^K \left(\frac{H_m}{H}\right)^a \binom{K+a}{a}}_{\textit{Probability mass function Pr(X=K)}} \underbrace{\left(1-\frac{H_m}{H}\right)(K+1-a)(1-f^c)R}_{\textit{Cost for each a}} + \underbrace{\frac{f^cR}{1-\beta}\frac{H_m}{H}}_{\textit{Opportunity cost}} \tag{11}$$

From Theorem 6 we may observe that cost of conducting a double spending attack is increasing in R and K. This is very intuitive because, the longer the escrow period (K) and the higher the reward (R), the costlier it is produce sufficient amount of proof-of-work.

3.4.3 Main Results

For $\frac{1}{2} < \frac{H_m}{H} \leq 1$ a double spending attack may be conducted without any direct costs and opportunity costs are strictly increasing in $\frac{H_m}{H}$ we therefore limit our focus to the nontrivial case where $0 < \frac{H_m}{H} \leq \frac{1}{2}$.

Theorem 7 *While*

$$(1-\beta)\left(2(K+1) - \frac{(K+2)\binom{2K+2}{K+2}}{2^{2K+1}}\right) \leq \frac{f^c}{1-f^c} \tag{12}$$

for all $0 < \frac{H_m}{H} \leq \frac{1}{2}$ *cost of double spending is larger than for* $0 = \frac{H_m}{H}$.

See Appendix D for a proof.

From Theorem 7 it follows that for reasonably small positive fees and high annual discount rates network concentration is not harmful for maintaining honest conduct. Quite contrary, larger pools are less likely to conduct a double spending attack. From Equation 12 we may observe that the threshold for f^c is increasing in K and decreasing in β. For an annual $\beta = 0.9$, average block intervals of 10 minutes and different $K = 6$, 60, 600 and 6000 the corresponding thresholds for fees are 0.003, 0.024, 0.24,and 2.3 percent. The current pool average fee in e.g. Bitcoin is over 2 percent. Obviously, this does not indicate that all of those fees are economic profits; however, in the case of the largest pool operator Bitmain, from January 2015 to June 2018, the gross profit margins have ranged between 80 to 89 percent [BitMain, 2018]. Even with rather conservative discount rates, limited economic profits are enough for the condition in Theorem 7 to hold.

The longer the escrow period K is, the larger the proof-of-work burden becomes for an attacker (for a detailed discussion about escrow times see [Chiu and Koeppl, 2019]). This affects directly the cost of double spending. One might argue that for a major attack, an attacker would need to conduct a large transaction and hence large transactions should have a long escrow period. However, to conduct a double spending attack one could conduct

multiple small transactions without being observed; and hence, we argue that K should correspond to the average time in a particular cryptocurrency protocol. Hence, e.g. for Bitcoin the most relevant escrow periods would be 6 to 60 blocks corresponding to average escrow times of 1 to 10 hours. In addition, it would be hard to argue that pools would benefit form the hashing power they control if the escrow periods where longer, because during an attack the pools would not be able to distribute rewards for the participants e.g. $K = 6000$ and 10 minute block intervals corresponds to an attack that would require on average 42 days. It is reasonable to assume that during those 42 days miners would exit the mining pool.

4 Conclusion

Mining pools' present value is dependent on the value of the network: Pool's future profits consist of collected fees which are proportional to the rewards paid in the cryptocurrency. These profits are hence dependent on the value of the cryptocurrency. By attacking against the network a pool would lose it's future profits through a collapse in the network's value. Therefore, even if large pools are more able to conduct double spending attacks, they are less willing to do so.

Concentration of mining to large pools in different blockchains has been considered a vital threat to trust and viability. This stems from the fact that in a concentrated network large pools can conduct double spending attacks more easily. If a pool controlled more than half of the network's hashing power, it would be able to use the same funds multiple times. Our analysis focuses on simple and intuitive economic incentives for large pools to maintain honest conduct. We derive thresholds of fees, discount rates and escrow periods for which large pools would be worse off by double spending, even though they are capable of conducting them. We conclude that, with conservative discount rates, and pool fees and escrow periods aligned with those observed in main crypto-currencies, the historically

observed pool concentration does not indicate a higher risk of double spending attacks. Hence, our result directly contradicts the common belief that concentration is harmful[14]. This result demonstrates the well-known economic insight that feasibility does not imply desirability.

[14]Our analysis and its results focus in a single aspect of this kind of attacks, mainly the purely economic incentives. There are other aspects such as political, ideological or environmental incentives that could be considered in order to gain a more holistic understanding of this phenomenon. The contribution of our analysis gives insight from one concrete perspective. Future work could consider other cases such as frictions in acquiring hash power and competition between different projects.

References

[Akerlof, 1970] Akerlof, G. A. (1970). The market for "lemons": Quality uncertainty and the market mechanism. *The Quarterly Journal of Economics*, 84(3):488–500.

[Bacharach and Bernasconi, 1997] Bacharach, M. and Bernasconi, M. (1997). The variable frame theory of focal points: An experimental study. *Games and Economic Behavior*, 19(1):1–45.

[Back et al., 2002] Back, A. et al. (2002). Hashcash-a denial of service counter-measure.

[Becker et al., 2012] Becker, J., Breuker, D., Heide, T., Holler, J., Peter Rauer, H., and Böhme, R. (2012). Can we afford integrity by proof-of-work? scenarios inspired by the bitcoin currency.

[Biais et al., 2018] Biais, B., Bisière, C., Bouvard, M., and Casamatta, C. (2018). The blockchain folk theorem. Swiss Finance Institute Working Paper 17-75, Institut d'Économie Industrielle (IDEI), Toulouse.

[BitMain, 2018] BitMain, T. H. C. (2018). Application proof of bitmain. Technical report.

[Böhme et al., 2015] Böhme, R., Christin, N., Edelman, B., and Moore, T. (2015). Bitcoin: Economics, technology, and governance. *Journal of Economic Perspectives*, 29(2):213–38.

[Bonneau et al., 2015] Bonneau, J., Miller, A., Clark, J., Narayanan, A., Kroll, J. A., and Felten, E. W. (2015). Sok: Research perspectives and challenges for bitcoin and cryptocurrencies. In *2015 IEEE Symposium on Security and Privacy*, pages 104–121. IEEE.

[Budish, 2018] Budish, E. (2018). The economic limits of bitcoin and the blockchain. Technical report, National Bureau of Economic Research.

[Chaum, 1982] Chaum, D. (1982). Blind signatures for untraceable payments (1983). In *Advances in Cryptology*.

[Chiu and Koeppl, 2019] Chiu, J. and Koeppl, T. V. (2019). Blockchain-based settlement for asset trading. *Review of Financial Studies*, 32(5):1716–1753.

[Cong et al., 2018] Cong, L., He, Z., and Li, J. (2018). Decentralized mining in centralized pools. *George Mason University School of Business Research Paper No. 18-9*.

[Cong and He, 2019] Cong, L. W. and He, Z. (2019). Blockchain disruption and smart contracts. *The Review of Financial Studies*, 32(5):1754–1797.

[Dimitri, 2017] Dimitri, N. (2017). Bitcoin mining as a contest. *Ledger*, 2:31–37.

[Dwork and Naor, 1992] Dwork, C. and Naor, M. (1992). Pricing via processing or combatting junk mail. In *Annual International Cryptology Conference*, pages 139–147. Springer.

[Easley et al., 2019] Easley, D., O'Hara, M., and Basu, S. (2019). From mining to markets: The evolution of bitcoin transaction fees. *Journal of Financial Economics*.

[Eyal and Gun Sirer, 2013] Eyal, I. and Gun Sirer, E. (2013). Majority is not enough: Bitcoin mining is vulnerable. volume 8437.

[Gans and Halaburda, 2015] Gans, J. S. and Halaburda, H. (2015). Some economics of private digital currency. In *Economic Analysis of the Digital Economy*, pages 257–276. University of Chicago Press.

[Gencer et al., 2018] Gencer, A. E., Basu, S., Eyal, I., Van Renesse, R., and Sirer, E. G. (2018). Decentralization in bitcoin and ethereum networks. *arXiv preprint arXiv:1801.03998*.

[Green and Porter, 1984] Green, E. J. and Porter, R. H. (1984). Noncooperative collusion under imperfect price information. *Econometrica: Journal of the Econometric Society*, pages 87–100.

[Haber and Stornetta, 1990] Haber, S. and Stornetta, W. S. (1990). How to time-stamp a digital document. In *Conference on the Theory and Application of Cryptography*, pages 437–455. Springer.

[Huang, 2003] Huang, G. T. (2003). The web's new currency. *Technology Review*, 106(10):28–28.

[Huberman et al., 2017] Huberman, G., Leshno, J., and Moallemi, C. C. (2017). Monopoly without a monopolist: An economic analysis of the bitcoin payment system. *Bank of Finland Research Discussion Papers*.

[Jakobsson and Juels, 1999] Jakobsson, M. and Juels, A. (1999). Proofs of work and bread pudding protocols. In *Secure Information Networks*, pages 258–272. Springer.

[Kaiser et al., 2018] Kaiser, B., Jurado, M., and Ledger, A. (2018). The looming threat of china: An analysis of chinese influence on bitcoin. *arXiv preprint arXiv:1810.02466*.

[Kiayias et al., 2016] Kiayias, A., Koutsoupias, E., Kyropoulou, M., and Tselekounis, Y. (2016). Blockchain mining games. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, EC '16, pages 365–382, New York, NY, USA. ACM.

[Kroll et al., 2013] Kroll, J. A., Davey, I. C., and Felten, E. W. (2013). The economics of bitcoin mining, or bitcoin in the presence of adversaries. In *Proceedings of WEIS*, volume 2013, page 11.

[Laurie, 2004] Laurie, B. (2004). Proof-of-work" proves not to work.

[Lee, 2016] Lee, L. (2016). New kids on the blockchain: How bitcoin's technology could reinvent the stock market. *Hastings Business Law Journal*, 12.

[Ma et al., 2018] Ma, J., Gans, J. S., and Tourky, R. (2018). Market structure in bitcoin mining. Technical report, National Bureau of Economic Research.

[Mehta et al., 1994] Mehta, J., Starmer, C., and Sugden, R. (1994). The nature of salience: An experimental investigation of pure coordination games. *The American Economic Review*, 84(3):658–673.

[Miller and LaViola Jr, 2014] Miller, A. and LaViola Jr, J. J. (2014). Anonymous byzantine consensus from moderately-hard puzzles: A model for bitcoin. *Available on line: http://nakamotoinstitute. org/research/anonymous-byzantine-consensus*.

[Modigliani and Miller, 1958] Modigliani, F. and Miller, M. H. (1958). The cost of capital, corporation finance and the theory of investment. *The American*, 1:3.

[Nakamoto, 2008] Nakamoto, S. (2008). Bitcoin: A peer-to-peer electronic cash system.

[Pagnotta and Buraschi, 2018] Pagnotta, E. and Buraschi, A. (2018). An equilibrium valuation of bitcoin and decentralized network assets. *Available at SSRN 3142022*.

[Pease et al., 1980] Pease, M., Shostak, R., and Lamport, L. (1980). Reaching agreement in the presence of faults. *J. ACM*, 27(2):228–234.

[Rosenfeld, 2011] Rosenfeld, M. (2011). Analysis of bitcoin pooled mining reward systems. *arXiv preprint arXiv:1112.4980*.

[Rosenfeld, 2014] Rosenfeld, M. (2014). Analysis of hashrate-based double spending. *arXiv preprint arXiv:1402.2009*.

[Rotemberg and Saloner, 1986] Rotemberg, J. and Saloner, G. (1986). A supergame-theoretic model of price wars during booms. *American Economic Review*, 76(3):390–407.

[Schelling, 1960] Schelling, T. C. (1960). *The strategy of conflict*. Harvard university press.

[Van Wirdum, 2018] Van Wirdum, A. (2018). How the bitcoin cash "hash war" came and went and not much happened.

[Yermack, 2017] Yermack, D. (2017). Corporate Governance and Blockchains*. *Review of Finance*, 21(1):7–31.

Appendix A. Main categories of mining pools by fee types

Mining pools use various fee and payment schemes. This section presents the most typical categories. A deeper presentation of the multiple payment schemes can be found in [Rosenfeld, 2011].

Pay-per-Share (PPS): the most basic insurance mechanism in pool mining. The pool rents hash power from the miners at fixed price, regardless of whether the pool is able to forge blocks or not. Therefore it transfers the risk to the pool managers. Typically, PPS pools have the highest fees.

Shared Maximum Pay Per Share (SMPPS): especial kind of PPS that limits the payment to the miners to the earnings of the pool.

Proportional: Similar to PPS, but rewards are shared only when the pool forges a block.

Pay Per Last N Shares (PPLNS): as the number of hashes used to forge a block varies from one block to another, PPLNS uses an especial kind of proportional fee setting contract. The rewards are distributed when the pool finds a block, but rather than paying by share of hashes divided by the total number of hashes needed for that block, it divides the share of hashes provided by the miner by a fixed number.

Slush's Bitcoin Pooled Mining (BMP) or Score: Introduced by Slush pool, uses a proportional scheme that weights shares to time during one mining round, so that later shares are rewarded more than early ones. The system was introduced to deincentivize pool switching during one round.

Geometric method (GM): the pool first takes a fixed fee from the block reward and distributes the rest among all miners in proportion to their score keeping the expected payoff per submitted share constant regardless of time during one round (mining one block) [Rosenfeld, 2011].

Double Geometric Method (DGM): a midway between GM and PPLNS. At every new block part of the score of the miners is transferred to the pool, so that if multiple blocks are found in a row, the pool keeps more rewards. On the other hand, as the time until forging a valid block increases, the miners keep a higher expected revenue. This method moves most of the risk to the pool, and thus is benefitial to it when the pool successfully finds multiple blocks, and benefitial for the risk-averse miners as their revenues get insurance against the pool not finding a block in a longer time [Rosenfeld, 2011].

Peer-to-Peer Mining Pool (P2Pool): similar to proportional pools but works in a decentralized peer-to-peer architecture. Miners in the pool work their own independent blockchain and, when finding a block that meets the difficulty criterion of the main chain, they merge it into the main chain and share the rewards proportionally to their share in their own independent blockchain. The goal of a P2Pool is to decentralize pool mining by substituting its central server with a P2P network owned by the members of the pool.

Appendix B. Glossary of Blockchain Terminology

51% Attack: A miner or group of miners that own more than half of the hashing power in the network and use it to generate an alternative chain that contains fraudulent blocks.

Altcoin: Alternative cryptocurrencies.

Application Specific Integrated Circuit (ASIC): A chip designed to complete very efficiently an specific task. The apparition of ASICs designed for Bitcoin mining in 2013 raised exponentially the total hash of the network and made mining with normal computers completely obsolete.

Bitcoin: The decentralized peer-to-peer cryptographic currency that introduced proof-of-work based blockchain technology. We use the term Bitcoin for the protocol, the network and the currency. The term bitcoin without capital letters refers to a unit of cryptocurrency. Bitcoin was introduced together with blockchain technology and thus was its first application.

Bitcoin Whitepaper: The paper "Bitcoin: A Peer-to-Peer Electronic Cash System" by Satoshi Nakamoto (2008) that introduced the concepts of Blockchain and of Bitcoin, together with its protocol. Available online here: bitcoin.org/bitcoin.pdf.

Block: A compilation of transactions, information, and a header. Blockheaders contain a reference to the previous block, which in turn refers to the previous one and so forth until the first block. This main structure of connected blocks gives name to the blockchain.

Block reward: Most blockchain protocols assign a reward when a miner adds a new block to the blockchain. Normally the reward is based on transaction fees and the creation of new units of the currency at every block . Rewards create incentives to mine and to keep the blockchain consistent. Bitcoin's protocol sets that the reward per block began at 50 BTC and is halved every 280 000 blocks. Once the total amount of bitcoins reaches 21 million, the block reward will be only the transaction fees decided by the users and no new bitcoins will be created.

Blockchain: The technology used to concatenate information in blocks and form a ledger or list of events. Blockchains can be private or public (centralized or decentralized). Blockchain technology was first proposed in Nakamoto's Whitepaper.

Byzantine Generals' Problem: The problem of gaining consensus in a network of parties where some might be faulty or fraudulent and there is no information about how to know which ones.

Cryptocurrency: A digital unit of value used as currency and based in the use of cryptography.

Cryptography: The science of securing a message and/or its parts through ciphers and codes.

Difficulty Level: The number of consequitive zeroes required in the output of the encription function used in a blockchain. Typically this number is adjusted dynamically as the amount of hash in the network changes to keep the expected time between blocks constant.

Double spending: Using the same units of cryptocurrency two times by substituting the block in the main chain where those cryptocurrencies were used. Double spending becomes virtually impossible after few blocks, unless the party that undertakes the fraud owns more than half of the hash power of the network.

Elliptic Curve Cryptography: The cryptographic algorithm used to generate Public Keys from Private Keys.

Ethereum: A public blockchain cryptocurrency focused in smart contracts that was introduced in 2015. Ethereum uses a proof-of-work algorithm called Ethash that aims to reduce the prominence of ASIC mining. In July, 2016 Ethereum had a hard fork, resulting in Ethereum and Ethereum Classic.

Fork: The division of a blockchain into two or more alternative chains. The forks create different tracks of transactions and thus distort ownership. Typically a fork will end when one of the chains becomes longer. However, both chains might continue to exist as independent blockchains, in which case is called a hard-fork. Prominent examples of hard forks are the split of Bitcoin into Bitcoin and Bitcoin Cash in August, 2017, and the split between Ethereum and Ethereum Classic in July, 2016.

Genesis Block: The first block of a blockchain. The Genesis Block is especial because it is the only block that doesn't refer to previous blocks.

Goldfinger attack: An attack that aims to disrupt a blockchain protocol or its consensus. The aim of the attack is not based in incentives related to gaining power or utility from the cryptocurrency but from damaging it.

Hash rate/hashing power: The amount of nonces (numbers) that a processor can generate per second, i.e. the processing power of a machine, when used to calculate hashing functions.

Initial Coin Offering: To offer initial units of a cryptocurrency in order to raise funds to establish it.

Miner: A node in a blockchain network that generates new blocks.

Mining: The action of forging new blocks by finding a nonce that, together with the rest of the blockheader, when used as input in the hashing function gives as output a number small enough to meet the difficulty criteria.

Mining Pool: A risk-sharing coalition of miners where each miner participates with its own hash and rewards are distributed following some payoff mechanism. Typically, a mining pool will charge a fee to participate and will either divide won rewards among its participants or will pay per hash, independently of wether a pool member forges the next block or not.

Node: A computer in a blockchain network.

Nonce: A random number that forms part of the content of a blockheader. The nonce is the only variable content in the block. It is changed so that the blockheader's content, when used as input in a hashing function, produces as output a number that meets the difficulty criteria.

Orphan Block: A properly mined block that ends in a discontinued fork.

Peer-to-Peer (P2P): A system of interconnected nodes that doesn't rely in a central coordinator such as a server or similar. Is widely used for file-sharing networks, cryptocurrencies and other applications. Typically P2P systems are open and based in decentralized cooperation.

Private Key: A secret code that is used for message ciphering. It is used together with the public key, such that public keys are widely known and used to encript a message that only the holder of the private key can read. In blockchains private keys are used to sign transactions.

Proof-of-stake (PoS): An alternative consensus protocol to proof-of-work. In PoS protocols the right to forge the next block is based on the stake of the node. PoS was developed mainly to reduce the energy externalities of proof-of-work protocols, however its effect on incentives is unclear and has rised critizism.

Proof-of-work: Used in the blockchain to ensure that creation of blocks is costly and thus, deter spam generation of blocks. Abbreviated as PoW, it is normally based in using

processing power to find the input to a hash function such that its output meets a difficulty criteria.

Public Key: A cryptographic key that refers to a person and allows to encrypt messages so that the owner of the key can decipher them using his private key.

Quantum computing attack: The use of quantum computers to hack a private key associated with a public key. This kind of attacks are still speculative, but might become more relevant as quantum computing technology develops. The use of quantum computers reduces drastically the number of trials the computer needs to discover a private key.

Satoshi: One hundred millionth of a bitcoin. Named after the author of the Bitcoin Whitepaper. One satoshi is the smallest fraction of a bitcoin accepted by the Bitcoin protocol.

SHA-256: A version of the Secure Hash Algorithm 2, whose output is a 256 bit string. It is the function used as PoW when mining bitcoins, so that the block-header is the input to the SHA-256 function and the output is the hash number of the block.

Transaction fee: An amount of currency linked to a transaction and that is assigned to the miner that includes the transaction into a validated block. Some cryptocurrencies require mandatory fees, while others -such as Bitcoin- rely on voluntary fees.

Appendix C. Difficulty Adjustment

To generate a new block a miner needs to guess a *nonce* which, with the other information in the block header, generates a hash number meeting a difficulty restriction. In proof-of-work protocols valid nonces are found by trying random numbers until one of them, together with the rest of the content of the block, generates via hash function a hash number that meets this criteria. This process requires computational power and, therefore, is costly. As an example, Bitcoin employs as PoW a difficulty criterion that dynamically sets a required count of consecutive zeros at the beginning of the hash number. The larger the number of

zeros required by the difficulty criterion, the harder it is to find a proper nonce.

The difficulty adjustment (at least in the major protocols) serves two purposes: First, it keeps the expected addition rate of new blocks constant (e.g. 10 minutes in Bitcoin, and 17 to 19 seconds in Ethereum). Hence, the time between blocks follows an exponential distribution, meaning that blocks are mined following a Poisson process at a constant average rate regardless of the network's total hash power. Second, if blocks would always require a fixed amount of hash power there could easily emerge disparities between cost of mining a block and a fixed reward granted for a block. There could emerge cases in which block reward is higher than the average cost of finding a nonce. Therefore, block creation would be accelerated, because new units of computational power would increase number of blocks mined but not diminish the reward per block. The fixed difficulty could also cause mining to be nonprofitable and hence to cease until, for exogenous reasons, price of computational power decreases or rewards appreciate. Hence, difficulty adjustment acts as a market clearing mechanism.

The difficulty level can be modeled in the following way. Let $i \in I = \{1, 2, ..., n\}$ be a node in the network, such that each node has an amount of hashing power [15] h_i. Following the characteristics of exponential distributions, node i's instantaneous probability of creating a block meeting the difficulty criterion is $x_i = \frac{h_i}{D}$, where D is the difficulty level. The protocol modifies regularly the difficulty level to keep the expected time T between blocks constant. We can define

$$T = \frac{1}{\sum_{i \in I} x_i} \tag{13}$$

[15] Hashing power is defined as the number of *nonces* per second a machine can calculate. It is related both to the machine's processing power and to the structure of its processor. Bitcoin and similar blockchain environments changed drastically with the introduction of Application-Specific Integrated Circuits (ASICs) designed specifically to maximize their hashing power's efficiency. ASICs are designed to perform only this task, and thus have substantially larger hashing power, typically ranging between 4 and 16 Terahash per second (TH/s). For comparison, an Intel Core i7 has a hashing power between 10 and 20 MH/s, i.e. around 1 million times less than an ASIC. The total hashrate of the Bitcoin network by the beginning of June 2018 was around 31 million TH/s.

Therefore,

$$T = \frac{1}{\sum_{i \in I} \frac{h_i}{D}} \tag{14}$$

The difficulty criterion is then defined by

$$D = T \sum_{i \in I} h_i \tag{15}$$

Hence, the difficulty increases as the total amount of hash $\sum_{i \in I} h_i$ increases; and miner i finds a *nonce* that meets the criterion following $\tilde{B} \sim Poisson(\frac{t}{D}\frac{h_i}{H})$. Therefore, without a loss of generality, in our model we can make a simplifying assumption that mining rewards are randomly assigned to a node (a miner or a pool) at every period. This allows us to avoid unnecessary complexity generated by explicitly modelling reward arrivals as Poisson process and difficulty adjustments.

Appendix D. Proofs

Proof of Corollary 4

Proof. From Assumption 3 it follows that an attacker must conclude the attack at some T. For a proof we need to demonstrate that while $B_{main} \geq K$ and $B_{attack} \leq B_{main}$ an attacker would be better off by paying $CH(B_{main,T-1}+1-B_{attack,T-1})(1-\frac{H_m}{H})$ with certainty than by paying $0 \leq \frac{H_a}{H}CH < (1-\frac{H_m}{H})CH$ with certainty and $CH(B_{main,T-1} - B_{attack,T-1})(1-\frac{H_m}{H})$ with probability $\frac{H_a+H_m}{H} < 1$ and $CH(B_{main,T-1} - B_{attack,T-1}+2)(1-\frac{H_m}{H})$ with probability $1-\frac{H_a+H_m}{H}$. If this holds, then by backward induction the attacker would be better off by acquiring all available hashing power immediately. These conditions yield the following inequality

$$
\begin{aligned}
&(B_{main,T-1}+1-B_{attack,T-1})\left(1-\frac{H_m}{H}\right)CH \leq \\
&\left\{\frac{H_a+H_m}{H}(B_{main,T-1}-B_{attack,T-1})\left(1-\frac{H_m}{H}\right)\right. \\
&\left.+\frac{H_a}{H}+\left(1-\frac{H_a+H_m}{H}\right)(B_{main,T-1}+2-B_{attack,T-1})\left(1-\frac{H_m}{H}\right)\right\}CH
\end{aligned} \tag{16}
$$

for $0 < \frac{H_a}{H} \leq (1-\frac{H_m}{H})$,

A simple manipulation yields

$$
0 < \left(1-\frac{H_m}{H}\right)\left(1-2\frac{H_m+H_a}{H}\right)+\frac{H_a}{H} \tag{17}
$$

Substituting $\frac{H_a}{H}$ with $1-\frac{H_m}{H}-\eta$ where $0 < \eta \leq 1-\frac{H_m}{H}$ yields

$$
0 \leq \left(1-2\frac{H_m}{H}\right)\eta \tag{18}
$$

which holds for $\frac{H_m}{H} < \frac{1}{2}$ as an inequality and for the special case $\frac{H_m}{H} = \frac{1}{2}$ as an equality. Implying that for $\frac{H_m}{H} < \frac{1}{2}$ an attacker strictly prefers ending the attack immediately and weakly for the special case.

■

Proof of Corollary 5

Proof. For the trivial case $B_{attack} > K$ it is self evident that $\frac{H_a}{H} > 0$ increases costs without any benefit.

To prove Corollary 5 it is necessary and sufficient to demonstrate that for all $B_{attack} \leq K$ and $B_{main} < K$ it holds that

$$
\frac{H_a+H_m}{H}C_W+\left(1-\frac{H_a+H_m}{H}\right)C_L+\frac{H_a}{H}CH > \frac{H_m}{H}C_W+\left(1-\frac{H_m}{H}\right)C_L \tag{19}
$$

where, C_W and C_L are expected cost of conducting a double spending attack after winning and losing one mining competition, respectively.

$$C_W = \sum_{a=0}^{K} \underbrace{\left(1 - \frac{H_m}{H}\right)^K \left(\frac{H_m}{H}\right)^a \binom{K+a}{a}}_{\text{Probability mass function}} \underbrace{\left(1 - \frac{H_m}{H}\right)(K+1-a-1)CH}_{\text{Cost for each a}} \tag{20}$$

$$C_L = \sum_{a=0}^{K} \underbrace{\left(1 - \frac{H_m}{H}\right)^{K-1} \left(\frac{H_m}{H}\right)^a \binom{K+a-1}{a}}_{\text{Probability mass function}} \underbrace{\left(1 - \frac{H_m}{H}\right)(K+1-a)CH}_{\text{Cost for each a}} \tag{21}$$

Both of these are sums of probability mass functions for each positive cost $(1 - \frac{H_m}{H})(K - a)CH$ and $(1 - \frac{H_m}{H})(K - a + 1)CH$, respectively, where a is a number of blocks mined by the attacker before $B_{main} = K$.

Substituting C_W and C_L to Equation 19, rearranging and manipulating yields

$$CH > C_L - C_W = 1 > \sum_{a=0}^{K} \left(1 - \frac{H_m}{H}\right)^{K-1} \left(\frac{H_m}{H}\right)^a \binom{K-1+a}{a} \left(1 - \left(\frac{H_m}{H}\right)^{K+1-a}\right) \tag{22}$$

which holds because

$$Pr(a \leq K) = \sum_{a=0}^{K} \left(1 - \frac{H_m}{H}\right)^{K-1} \left(\frac{H_m}{H}\right)^a \binom{K-1+a}{a} \leq 1 \tag{23}$$

and, for any value $K + 1 - a < \infty$. [16]

$$\left(1 - \left(\frac{H_m}{H}\right)^{K+1-a}\right) < 1 \tag{24}$$

holds.

■

[16] $K+1-a = \infty$ would imply infinite cost for conducting a double spending attack and an infinite waiting time for goods to be delivered.

Proof of Theorem 7

Proof.

For cost of double spending to be larger for $0 < \frac{H_m}{H} \leq \frac{1}{2}$ than for $0 = \frac{H_m}{H}$ it must be the case that

$$(1-f^c)R\left\{(K+1) - \frac{\left(1-2\frac{H_m}{H}\right)(K+1) + \binom{2K+2}{K+2}((1-\frac{H_m}{H})\frac{H_m}{H})^{K+2}\,{}_2F_1(2,2K+3;K+3,\frac{H_m}{H})}{1-\frac{H_m}{H}}\right\} - \frac{f^cR}{1-\beta}\frac{H_m}{H} < 0 \tag{25}$$

For every $0 < \frac{H_m}{H} \leq \frac{1}{2}$ and $0 < K$. Where $(1-f^c)R(K+1)$ is the cost of double spending for an attacker with $\frac{H_m}{H} = 0$.

Simple rearrangement of Equation 25 yields

$$(1-f^c)\left\{(K+1) - \frac{(1-2\frac{H_m}{H})(K+1) + \binom{2K+2}{K+2}((1-\frac{H_m}{H})\frac{H_m}{H})^{K+2}\,{}_2F_1(2,2K+3;K+3,\frac{H_m}{H})}{1-\frac{H_m}{H}}\right\}\frac{H}{H_m} < \frac{f^c}{1-\beta} \tag{26}$$

Taking a partial derivative w.r.t. $\frac{H_m}{H}$ of the left hand side of the Equation 24 and setting it to zero yields $\frac{H_m}{H} = \frac{1}{2}$. This and $f''(\frac{1}{2}) < 0$ imply that it is sufficient to analyze inequality at $\frac{H_m}{H} = \frac{1}{2}$; i.e. where the direct benefit from H_m is largest in comparison to opportunity costs. After analysing the inequality 26 in a case where $\frac{H_m}{H} = \frac{1}{2}$ and applying Gauss' Summation Theorem to the hypergeometric function and Legendre's relation and some manipulation and expressing hypergeometric function as a sum Equation 24 simplifies to the equation in Theorem 7. ■